Theoretical Computer Science 410 (2009) 2489–2510

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Parallelizing quantum circuits

Anne Broadbent [a,*], Elham Kashefi [b]

[a] *Département d'informatique et de recherche opérationnelle, Université de Montréal, Canada*

[b] *University of Edinburgh, School of Informatics, Laboratoire d'Informatique de Grenoble, United Kingdom*

ARTICLE INFO

Article history:
Received 11 October 2007
Received in revised form 18 July 2008
Accepted 23 December 2008
Communicated by S. Hallgren

Keywords:
Quantum circuits
Circuit depth
Measurement-based quantum computing
Measurement calculus

ABSTRACT

We present a novel automated technique for parallelizing quantum circuits via the forward and backward translation to measurement-based quantum computing patterns, and analyze the trade off in terms of depth and space complexity. As a result we distinguish a class of polynomial depth circuits that can be parallelized to logarithmic depth while adding only a polynomial number of auxiliary qubits. In particular, we provide for the first time a full characterization of patterns with flow of arbitrary depth, based on the notion of influencing walks and a simple rewriting system on the angles of the measurement. Our method provides new insight for constructing parallel circuits and as applications, we demonstrate several classes of circuits that can be parallelized to constant or logarithmic depth. Furthermore, we prove a logarithmic separation in terms of quantum depth between the quantum circuit model and the measurement-based model.

1. Introduction and summary of results

The development of small depth quantum circuits seems almost essential if we wish to implement quantum algorithms in the near future with the available technology. Due to decoherence, qubits have a tendency to spontaneously change their state, hence we can only operate on them for a very short period of time. The parallelization of circuits could maximize the use of these fragile qubits. Note that to obtain parallelism in the quantum circuit model, we need the ability to interact with spatially apart qubits. Different implementations might put physical limitations on how far we can apply this ability. However, in some recent proposals for quantum computing [1–7], the far-apart interaction between qubits has been successfully demonstrated.

From a theoretical point, the study of parallel quantum algorithms could lead to new results in complexity theory. For instance, one interesting open question is whether the class of decision problems solvable in polynomial time, **P**, is included in the class of decision problems solvable in polylogarithmic depth, **NC**. Let **QNC** be the class of decision problems solvable in polylogarithmic depth with a quantum computer, one can ask similarly whether **P** is included in **QNC**. Finally, Richard Jozsa conjectured that:

Jozsa Conjecture [8]. *Any polynomial-time quantum algorithm can be implemented with only* $O(\log(n))$ *quantum layers interspersed with polynomial-time classical computations.*

In other words all efficient quantum algorithms are believed to be parallelizable, this important conjecture is based on recent results in the formalism of the measurement-based model for quantum computation [8]. If the conjecture is true, then we might be able to prove that parallelism is an essential property of entanglement.

Previous results on parallel quantum circuits include the parallelization of circuits for the semi-classical quantum Fourier transform [9], approximate quantum Fourier transform [10], as well as for encoding and decoding quantum error-correcting

* Corresponding address: Institute for Quantum Computing, University of Waterloo, Canada. Tel.: +1 519 514 2048.
E-mail addresses: broadbea@iro.umontreal.ca (A. Broadbent), ekashefi@inf.ed.ac.uk (E. Kashefi).

0304-3975/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2008.12.046

codes [11]. These constructions usually require the use of auxiliary qubits. The depth complexity of quantum circuits has also been studied in [12,13]. Several other approaches based on local optimization and circuit rewriting rules were introduced in [14,15].

Our results

We present a construction for the parallelization of quantum circuits where our method gives a formula that computes the exact decrease in depth that the construction can achieve. This yields a novel way for the construction of lower-depth quantum circuits. We will consider the universal set of gates $\mathcal{U}$ consisting of controlled-Z, $J(\alpha)$, H and H^i:

$$\wedge Z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad J(\alpha) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix}, \quad H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad H^i = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -i \\ 1 & i \end{pmatrix}.$$

The notion of *circuit influencing path* is the key concept in our automated parallelization techniques: a left-to-right path starting at the beginning of a circuit wire ending at the same or another wire, such that the jumps between wires are done through controlled-Z gates with no two consecutive jumps (see Section 5.3 for more explanation and example). Given an influencing path, we define the following simplification rule for the $J(\alpha)$ gates (for simplicity we omit the α parameter of the J gates):

$$J\ \mathsf{P}_1\ A_1B_1\ \mathsf{P}_2\ A_2B_2\ \cdots\ \mathsf{P}_k\ J \Rightarrow \begin{cases} J & \text{if } \exists \mathsf{P}_i = (H)^{\text{odd}}(H^i(H)^{\text{odd}})^* \\ JJ & \text{otherwise} \end{cases}$$

where P_i represents a finite sequence of H and H^i gates; $^{\text{odd}}$ represents an odd number of repetitions, and A_i and B_i represents the J gates immediately after a controlled-Z gate on the circuit influencing path (on the control and the target wires).

Theorem. *Let C be a circuit of gates in $\mathcal{U}$ on n qubits with size s and depth D. Assume that after the above simplification rule over all circuit influencing paths, we obtain at most D' many consecutive J gates. Then circuit C can be parallelized with an equivalent circuit C' with depth in $O(D' \log(s))$ and size in $O(s^3 + n)$.*

In simple words, the theorem states that the longest sequence of consecutive J gates over an influencing path is an upper bound of the circuit depth. However the "magical" sequence of $(H)^{\text{odd}}(H^i(H)^{\text{odd}})^*$, separating two J gates will make them appear in the same layer after parallelization is performed. Furthermore, this sequence is a constructive building block for designing parallel circuits as we discuss later. It is important to emphasize that the simplification rule is not a circuit identity and hence our parallelization method is fundamentally different from the local circuit rewriting approaches. We use influencing paths as a structural tool for analyzing circuit depth and then, using an automated method (described below) we can construct another circuit having the computed improved depth.

Our main theorem, the concept of the influencing path and the automated parallelization technique, are all obtained using the formalism of the measurement-based model for quantum computation (MBQC) [8,16–18]. In high level, we start with a circuit, translate it into a computation in MBQC, then we perform some depth-reducing operations and last translate it back into a circuit. Before describing our technique in more detail, let us define MBQC. A computation in MBQC is usually referred to as a *pattern* and consists of a round of global operations (two-qubit gates) to create the required initial multi-qubit entanglement, followed by a sequence of classically controlled local operators (single-qubit measurements and unitaries). A more formal definition is given later. We will work in particular within an algebraic framework for MBQC called the *measurement calculus* [19]. This novel framework is universal and equivalent in computational power to the quantum circuit model.[1] Previous results on the parallelization in the MBQC include constant-depth patterns for Clifford unitaries [20] and diagonal unitaries [18].

The measurement calculus framework clearly distinguishes between the quantum and classical depths of a pattern. Informally, the quantum depth of a pattern is the length of the longest sequence of dependent commands. The classical depth is the depth of the classical computation required for the evaluation of the dependency function of each dependent command. We consider two transformations that we can apply to patterns without changing their meaning (the underlying operator that they implement) while never increasing their depth (and possibly decreasing it): standardization (Theorem 4.2) and signal shifting (Theorem 4.3). Standardization is a rewriting system for MBQC patterns that pushes all the entanglement operators to the beginning of the computation, followed by a sequence of the single-qubit measurements and a final round of local unitaries. Signal shifting is another rewriting system that translates some of the quantum depth between measurement operators to classical depth between the final local unitaries and hence decreases the quantum depth.

We then develop a method to compute an upper bound on the quantum depth of a pattern. In order to do so, we use the notion of *flow* [21], a graph theoretical tool defined over the underlying geometry of the initial entanglement state of a

[1] In this paper whenever we mention a quantum circuit or a pattern we mean a uniform family of quantum circuits or patterns, where their descriptions are given by a classical Turing machine.

pattern. We further define a construction called an *influencing walk*, that allows us to characterize the dependency structure of the pattern. It is known that a particular set of measurements called Pauli measurements can be performed independently as the first layer of measurements [16]. Combining this fact about the angles of the measurement with influencing walks and the signal shifting procedure, we present an upper bound result on the quantum depth (Proposition 4.6). As for the classical depth, it is known to be at most logarithmic in the size of the pattern [8]. We give some tighter upper bounds based on the underlying geometry (Proposition 4.4).

Our ultimate goal is to decrease the depth of a given circuit, to this end we present an automated procedure for the translation of a circuit (with n gates) to an MBQC pattern by adding only up to n extra auxiliary qubits. Performing standardization and signal shifting over the obtained pattern might decrease the depth, and we then translate back the obtained low depth pattern to another circuit, equivalent to the original circuit but with lower quantum depth and more auxiliary qubits. This final translation is based on performing coherent measurements, and therefore the new circuit will have a depth equal to the combined quantum and classical depths of the pattern. Note that since classical computation is cheaper than quantum computation, one might consider MBQC as a favorable ultimate architecture for a quantum computer as it keeps the quantum and classical depth separate. However, this translation forward and backward to MBQC is interesting from the theoretical point of view as one can parallelize a circuit automatically and, moreover, due to the simplicity of the translation procedure, the pattern depth characterization of Theorem 4.8 leads to a general parallelization result for circuits, Theorem 5.12.

As already noted, the depth of a pattern is due to the adaptive measurements and corrections: any given qubit has a fixed set of measurement outcomes that must be known before a measurement or a correction command can be performed at that qubit. This set of measurement dependencies is sometimes called the *backward cone* [20]. One way of interpreting our main result given by Theorem 4.8 is that we characterize the backward cone of any qubit; thus for patterns with flow, we are able to give a method to easily compute the depth. Moreover our characterization result is constructive and leads to a novel technique for building parallel patterns and parallel circuits.

In order to demonstrate the power of Theorems 4.8 and 5.12, we present some special cases: depth 2 patterns (Proposition 6.1) and depth 2 circuits (Proposition 6.2). Another application of our results is the parallelization of the Clifford operators. Consequently, using the example of *parity function*, we show for the first time a logarithmic separation in terms of quantum depth between the circuit model and the MBQC. Finally, we show how our method can be used to parallelize a family of polynomial-depth circuits to logarithmic depth.

The paper is organized as follows. In Section 2, we briefly review the MBQC in order to fix the relevant notation. In Section 3, we define the notion of depth for a pattern in the MBQC, carefully distinguishing between the preparation, quantum and classical computation depths. In Section 4, we present our main techniques for reducing a pattern's depth: standardization, signal shifting and Pauli resetting; we also prove our main pattern depth characterization result. In Section 5, we give a translation from the quantum circuit model to the measurement-based model and back and present the circuit depth characterization result. Several applications of our results are given in Section 6.

2. Preliminaries

2.1. Quantum circuit model

Richard Feynman was one of the first to suggest that a computer based on the principles of quantum mechanics could efficiently *simulate* other quantum systems [22]. David Deutsch then developed the idea that the quantum computer could offer a computational advantage compared to the classical computer; he also defined the *quantum Turing machine* [23], before defining the *quantum circuit model* [24] to represent quantum computations (Deutsch refers to a quantum circuit as a quantum *network*). It is readily seen that the quantum circuit model is a generalization of the classical circuit model. We briefly review the required concepts from quantum computing, a more detailed introduction can be found in [25].

Let $\mathcal{H}$ denote a 2-dimensional complex vector space, equipped with the standard inner product. We pick an orthonormal basis for this space, label the two basis vectors $|0\rangle$ and $|1\rangle$, and for simplicity identify them with the vectors $\begin{pmatrix}1\\0\end{pmatrix}$ and $\begin{pmatrix}0\\1\end{pmatrix}$, respectively. A *qubit* is a unit length vector in this space, and so can be expressed as a linear combination of the basis states:

$$\alpha_0|0\rangle + \alpha_1|1\rangle = \begin{pmatrix}\alpha_0\\\alpha_1\end{pmatrix}.$$

Here α_0, α_1 are complex *amplitudes*, and $|\alpha_0|^2 + |\alpha_1|^2 = 1$.

An *m-qubit state* is a unit vector in the m-fold tensor space $\mathcal{H} \otimes \cdots \otimes \mathcal{H}$. The 2^m basis states of this space are the m-fold tensor products of the states $|0\rangle$ and $|1\rangle$. We abbreviate $|1\rangle \otimes |0\rangle$ to $|1\rangle|0\rangle$ or $|10\rangle$. With these basis states, an m-qubit state $|\phi\rangle$ is a 2^m-dimensional complex unit vector

$$|\phi\rangle = \sum_{i\in\{0,1\}^m} \alpha_i|i\rangle.$$

There exist quantum states that cannot be written as the tensor product of other quantum states, e.g. $|00\rangle + |11\rangle$. This means that given a general element of $\mathcal{H} \otimes \mathcal{H}'$ one cannot produce elements of $\mathcal{H}$ and $\mathcal{H}'$; such states are called *entangled* states.

We use $\langle\phi| = |\phi\rangle^*$ to denote the conjugate transpose of the vector $|\phi\rangle$, and $\langle\phi \mid \psi\rangle = \langle\phi| \cdot |\psi\rangle$ for the inner product between states $|\phi\rangle$ and $|\psi\rangle$. These two states are *orthogonal* if $\langle\phi \mid \psi\rangle = 0$. The *norm* of $|\phi\rangle$ is $\|\phi\| = \sqrt{|\langle\phi \mid \phi\rangle|}$.

A quantum state can evolve by a unitary operation or by a measurement. A *unitary* transformation is a linear mapping that preserves the norm of the states. If we apply a unitary U to a state $|\phi\rangle$, it evolves to $U|\phi\rangle$.

The *Pauli operators* are a well-known set of unitary transformations for quantum computing:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

and the *Pauli group* on n qubits is generated by Pauli operators. We introduced before several other unitary transformations here are two more, the identity I, the *phase* gate $Z(\alpha)$, of which $Z(\pi/4)$ and $Z(\pi/2)$ are a special cases:

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad Z(\alpha) = \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{i\alpha} \end{pmatrix}.$$

The *Clifford group* on n qubits is generated by the following matrices: $Z, H, Z(\pi/2)$ and $\wedge Z$. This set of matrices is not universal for quantum computation, but by adding any single-qubit gate not in the Clifford group (such as $Z(\pi/4)$), we do get a set that is approximately universal for quantum computing. The importance of the Clifford group for quantum computation is that a computation consisting of only Clifford operations on the computational basis followed by final Pauli measurements can be efficiently simulated by a classical computer, this is the Gottesman–Knill theorem [26,27].

The most general measurement allowed by quantum mechanics is specified by a family of positive semidefinite operators $E_i = M_i^* M_i$, $1 \leq i \leq k$, subject to the condition that $\sum_i E_i = I$. A projective measurement is defined in the special case where the operators are projections. Let $|\phi\rangle$ be an m-qubit state and $\mathcal{B} = \{|b_1\rangle, \ldots, |b_{2^m}\rangle\}$ an orthonormal basis of the m-qubit space. A projective measurement of the state $|\phi\rangle$ in the $\mathcal{B}$ basis means that we apply the projection operators $P_i = |b_i\rangle\langle b_i|$ to $|\phi\rangle$. The resulting quantum state is $|b_i\rangle$ with probability $p_i = |\langle\phi \mid b_i\rangle|^2$. An important class of projective measurements are Pauli measurements, i.e. projections to eigenstates of Pauli operators.

Any unitary operation U can be approximated with a circuit C, using gates in a fixed universal set of gates, e.g. controlled-Z and $J(\alpha)$ [28]. The *size* of a circuit is the number of gates and its *depth* is the largest number of gates on any input–output path. Equivalently, the depth is the number of layers that are required for the parallel execution of the circuit, where a qubit can be involved in at most one interaction per layer. In this paper, we adopt the model according to which at any given timestep, a single qubit can be involved in at most one interaction. This differs from the *concurrency* viewpoint, according to which all interactions for commuting operations can be done simultaneously.

2.2. Measurement-based model

We give a brief introduction to the MBQC a more detailed description is available in [8,17–19]. Our notation follows that of [19].

The measurement-based model [16,20,29] is a relatively new approach to quantum computation that is oriented around single-qubit measurements and entanglement for performing quantum computations. The model contrasts with the widely-used quantum circuit model where measurements usually occur only at the end of the circuit, their sole purpose being to obtain a classical output out of the quantum output. In measurement-based model, computations are represented as *patterns*, which are sequences of *commands* acting on the qubits in the pattern. These commands are of four types:

1. N_i is a one-qubit preparation command which prepares the auxiliary qubit i in state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. The preparation commands can be implicit from the pattern: when not specified, all non-input qubits are prepared in the $|+\rangle$ state.
2. E_{ij} is a two-qubit entanglement command which applies the controlled-Z operation, $\wedge Z$, to qubits i and j. Note that the $\wedge Z$ operation is symmetric and so $E_{ij} = E_{ji}$. Also, E_{ij} commutes with E_{jk} and so the ordering of the entanglement commands in not important.
3. M_i^α is a one-qubit measurement on qubit i which depends on parameter $\alpha \in [0, 2\pi)$ called the *angle of measurement*. M_i^α is the orthogonal projection onto states

$$|+_\alpha\rangle = \frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{i\alpha}|1\rangle)$$
$$|-_\alpha\rangle = \frac{1}{\sqrt{2}}(|0\rangle - \mathrm{e}^{i\alpha}|1\rangle),$$

   followed by a trace-out operator, since measurements are destructive. We denote the classical outcome of a measurement done at qubit i by $s_i \in \mathbb{Z}_2$. We take the specific convention that $s_i = 0$ if the measurement outcome is $|+_\alpha\rangle$, and that $s_i = 1$ if the measurement outcome is $|-_\alpha\rangle$. Outcomes can be summed together resulting in expressions of the form

$$s = \sum_{i \in I} s_i$$

   which are called *signals*, and where the summation is understood as being done modulo 2. The *domain* of a signal is the set of qubits on which it depends (in this example, the domain of s is I).
4. X_i and Z_i are one-qubit Pauli corrections which correspond to the application of the Pauli X and Z matrices, respectively, on qubit i.

In order to obtain universality, we have to add a classical control mechanism called *feed-forward*, which allows measurements and corrections to be dependent on the results of previous measurements [16,19]. Let s and t be signals. Dependent corrections are written as X_i^s and Z_i^s and dependent measurements are written as ${}_t[M_i^\alpha]^s$. The meaning of dependencies for corrections is straightforward: $X_i^0 = Z_i^0 = I$ (no correction is applied), while $X_i^1 = X_i$ and $Z_i^1 = Z_i$. In the case of dependent measurements, the measurement angle depends on s, t and α as follows:

$$ {}_t[M_i^\alpha]^s = M_i^{(-1)^s\alpha + t\pi} \tag{1}$$

so that, depending on the parity of s and t, one may have to modify the angle of measurement α to one of $-\alpha$, $\alpha + \pi$ and $-\alpha + \pi$. These modifications correspond to conjugations of measurements under X and Z:

$$X_i^s M_i^\alpha X_i^s = M_i^{(-1)^s\alpha} \tag{2}$$
$$Z_i^t M_i^\alpha Z_i^t = M_i^{\alpha + t\pi} \tag{3}$$

and so we will refer to them as the X- and Z-actions or alternatively as the X- and Z-dependencies. Since measurements are destructive, the above equations simplify to:

$$M_i^\alpha X_i^s = M_i^{(-1)^s\alpha} \tag{4}$$
$$M_i^\alpha Z_i^t = M_i^{\alpha + t\pi}. \tag{5}$$

Note that these two actions are commuting, since $-\alpha + \pi = -\alpha - \pi$ up to 2π, and hence the order in which one applies them does not matter.

A *pattern* is defined by the choice of a finite set V of qubits, two not necessarily disjoint sets $I \subseteq V$ and $O \subseteq V$ determining the pattern inputs and outputs, and a finite sequence of commands acting on V. We require that no command depend on an outcome not yet measured, that no command act on a qubit already measured, that a qubit be measured if and only if it is not an output qubit and that a qubit be prepared if and only if it is not an input qubit. This set of rules is known as the *definiteness* condition.

Just as circuits, patterns operate on a fixed number of input qubits. Such models of computation are called *non-uniform*. If we want to solve problems that are defined for an arbitrary input length, we need to construct one pattern for each length. This pattern family is an *infinite* object. By imposing some *uniformity conditions*, we require that the patterns for different input lengths have something in common concerning their structure. This, in turn, ensures that a pattern family has a finite description. These uniformity conditions are similar to those that are usually imposed on uniform families of *circuits* [30].

A pattern is said to be in *standard form* if all the preparation N_i and entanglement operators E_{ij} appear first in its command sequence, followed by measurements and finally corrections. A pattern that is not in standard form is called a *wild pattern*. Any wild pattern can be put in its unique standard form [19]; this form can reveal implicit parallelism in the computation, and is well-suited for certain implementations (see Section 4.1). The procedure of rewriting a pattern to its standard form is called *standardization*. This can be done by applying the following rewrite rules:

$$E_{ij}X_i^s \Rightarrow X_i^s Z_j^s E_{ij} \tag{6}$$
$$E_{ij}Z_i^s \Rightarrow Z_i^s E_{ij} \tag{7}$$
$${}_t[M_i^\alpha]^s X_i^r \Rightarrow {}_t[M_i^\alpha]^{s+r} \tag{8}$$
$${}_t[M_i^\alpha]^s Z_i^r \Rightarrow {}_{r+t}[M_i^\alpha]^s. \tag{9}$$

The rewrite rules also contain the following *free commutation rules* which tell us that, if we are dealing with disjoint sets of target qubits, measurement, corrections and entanglement commands commute pairwise [19].

$$E_{ij}A_{\vec{k}} \Rightarrow A_{\vec{k}}E_{ij} \quad \text{where } A \text{ is not an entanglement} \tag{10}$$
$$A_{\vec{k}}X_i^s \Rightarrow X_i^s A_{\vec{k}} \quad \text{where } A \text{ is not a correction} \tag{11}$$
$$A_{\vec{k}}Z_i^s \Rightarrow Z_i^s A_{\vec{k}} \quad \text{where } A \text{ is not a correction} \tag{12}$$

where $\vec{k}$ represent the qubits acted upon by command A, and are distinct from i and j. Clearly these rules could be reversed since they hold as equations but we are orienting them this way in order to obtain termination for the standardization procedure.

Under rewriting, the computation space, inputs and outputs remain the same, and so do the entanglement commands. Measurements might be modified, but we still measure exactly the same qubits. The only major modifications concern local corrections and dependencies. If there were no dependencies at the start, none would be created in the rewriting process.

Finally the last set of rewrite rules called *signal shifting rules*, allows us to dispose of dependencies induced by the Z-action, and obtain sometimes standard patterns with smaller depth complexity (see Section 4.2). We refer to signal shifting as the procedure of applying the signal shifting rules until no further rules are applicable:

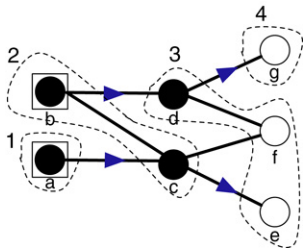

Fig. 1. A geometry with flow. The boxed vertices are the input qubits and the white vertices are the output qubits. All the non-output qubits, black vertices, are measured during the run of the pattern. The flow function is represented as arcs and the partial order on the vertices is given by the 4 partition sets.

$$ {}_t[M_i^\alpha]^s \Rightarrow S_i^t\ [M_i^\alpha]^s \tag{13} $$

$$ X_j^s S_i^t \Rightarrow S_i^t\ X_j^{s[(t+s_i)/s_i]} \tag{14} $$

$$ Z_j^s S_i^t \Rightarrow S_i^t\ Z_j^{s[(t+s_i)/s_i]} \tag{15} $$

$$ {}_t[M_j^\alpha]^s S_i^r \Rightarrow S_i^r\ {}_{t[(r+s_i)/s_i]}[M_j^\alpha]^{s[(r+s_i)/s_i]} \tag{16} $$

where S_i^t is the signal shifting command (adding t to s_i) and $s[t/s_i]$ denotes the substitution of s_i with t in s.

The execution of a pattern consists in performing each command in sequence, from right to left. If n is the number of measurements (i.e. the number of non-output qubits), then this may follow 2^n different computational branches. Each branch is associated with a unique binary string s of length n, representing the classical outcomes of the measurements along that branch, and a unique *branch map* A_s representing the linear transformation from the input Hilbert space to the output Hilbert space, along that branch. A pattern is said to be *deterministic* if all the branch maps are proportional, it is said to be *strongly deterministic* when branch maps are equal (up to a global phase), and it is said to be *uniformly deterministic* if it is deterministic for any choice of measurement angles. The following notions are beneficial for the study of dependency structure and determinism. A *geometry* (G, I, O) consists of an undirected graph G together with two subsets of nodes I and O, called inputs and outputs. We write V for the set of vertices in G, E for the set of edges, I^c, and O^c for the complements of I and O in V and $E_G := \prod_{\{i,j\}\in E} E_{ij}$ for the global entanglement operator associated to G (the graph G is also called the *entanglement graph* [31]). Trivially, any pattern has a unique underlying geometry, obtained by forgetting measurements and correction commands.

We now give a condition on geometries under which it is possible to synthesize a set of dependent corrections such that the obtained pattern is uniformly and strongly deterministic, i.e. all the branches of the computation are equal, independently of the angles of the measurements. Hence we obtain the dependency structure of measurement commands directly from the geometry, from which we will get a unified treatment of depth complexity for measurement patterns. In what follows, $x \sim y$ denotes that x is adjacent to y in G, N_{I^c} denotes the sequence of preparation commands $\prod_{i\in I^c} N_i$.

Definition 2.1 (*[21]*). A *flow* $(f, \preceq)$ for a geometry (G, I, O) consists of a map $f : O^c \to I^c$ and a partial order $\preceq$ over V such that for all $x \in O^c$:

(i) $x \sim f(x)$;
(ii) $x \preceq f(x)$;
(iii) for all $y \sim f(x), x \preceq y$.

Fig. 1 shows a geometry together with a flow, where f is represented by arcs from O^c (measured qubits, black vertices) to I^c (prepared qubits, non boxed vertices). The associated partial order is given by the labeled sets of vertices. The coarsest order $\preceq$ for which $(f, \preceq)$ is a flow is called the *dependency order* induced by f and its depth (4 in Fig. 1) is called *flow depth*.

Theorem 2.2 (*Flow theorem [21]*). *Suppose the geometry (G, I, O) has flow f, then the pattern:*

$$ \mathcal{P}_{f,G,\vec{\alpha}} := \prod_{i\in O^c}^{\preceq} \left(X_{f(i)}^{s_i} \prod_{\substack{k\sim f(i)\\ k\neq i}} Z_k^{s_i} M_i^{\alpha_i} \right) E_G $$

where the product follows the dependency order $\preceq$ of f, is uniformly and strongly deterministic, and realizes the unitary embedding:

$$ U_{G,I,O,\vec{\alpha}} := 2^{|O^c|/2} \left(\prod_{i\in O^c} \langle +_{\alpha_i}|_i \right) E_G. $$

The flow theorem (Theorem 2.2) plays an important role in our discussion of depth complexity in the following sections. If the underlying geometry of a pattern has a flow and its pattern command sequence is constructed as given by the flow theorem, we call this pattern a *pattern with flow*. Note that the flow theorem tells us how to perform dependent corrections according to the flow function f: when qubit i is measured, its neighbor according to the flow, $f(i)$, receives the $X_{f(i)}^{s_i}$ correction, while all the neighbors k of $f(i)$ (independently of the flow and except i), receive a $Z_k^{s_i}$ correction. We can apply the rewrite rules of Eqs. (8) and (9) to propagate these dependent corrections to the end and obtain a standard form for the pattern with flow:

$$\prod_{i\in O} X_i^{s_{f^{-1}(i)}} Z_i^{\sum_{j:f(j)\sim i} s_j} \prod_{i\in O^c}^{\preceq} {}_{\sum_{j:f(j)\sim i} s_j}[M_i^{\alpha_i}]^{s_{f^{-1}(i)}} E_G \tag{17}$$

where $f^{-1}(i)$ is well defined since by construction f is an one-to-one function. If $f^{-1}(i)$ is empty we ignore the term $s_{f^{-1}(i)}$, that means the measurement at qubit i has no X-dependency.

Given a geometry on n vertices with $|I| = |O|$, one can efficiently i.e. in $O(\mathbf{poly}(n))$ time, find its unique flow if it exists [32,33] and the obtained pattern implements a unitary operator.

3. Depth complexity for patterns

In this section, we give a definition for the preparation depth and give its exact value. We also give a definition for the quantum computation depth of a pattern, the analogue of the quantum circuit depth. One issue that has often been overlooked in the literature on MBQC is that computation of the correction exponents as well as the measurement angles contributes to a *classical* depth of patterns [8]. We will present an upper bound for the classical depth of patterns with flow.

First, we focus on the notion of depth complexity for a standard pattern, which extends naturally to wild patterns. There are two parts of a standard pattern computation that contribute to its depth: the *preparation* phase, which is the work required to prepare the entangled state (the N and E commands), and the *quantum computation* phase, which is the work required to perform the measurements and corrections (the adaptive M and C parts). The total depth of a pattern in standard form is the sum of the depths of the preparation and computation parts, which we address now separately.

3.1. Preparation depth

As already mentioned, for any pattern $\mathcal{P}$ with computational space (V, I, O) one can associate an underlying geometry (G, I, O) defined by forgetting the measurement and correction commands. The entangled state corresponding to this geometry is defined by preparing the input qubits in the given arbitrary states and all other qubits in the $|+\rangle$ state and applying a $\wedge Z$ on all qubits i and j that are adjacent in the entanglement graph G. We give below an exact value for this depth, in terms of $\Delta(G)$, the maximum degree of G. A similar result also appeared in [34].

Lemma 3.1. *The preparation depth for a given entanglement graph G, is either $\Delta(G)$ or $\Delta(G)+1$.*

Proof. At each timestep, a given qubit can interact with at most one other qubit. In terms of the entanglement graph, this means that at each timestep, a given node can interact with at most one of its neighbors. Assign a color to each timestep and color the edge in the entanglement graph G accordingly. With this view, the entire preparation corresponds to an edge coloring of the entanglement graph. By Vizing's theorem [35], the edge-chromatic number of G, $\chi'(G)$ satisfies $\Delta(G) \leq \chi'(G) \leq \Delta(G)+1$. □

It is known that a special type of entanglement graph, the two-dimensional grid (called *cluster state*), is universal for the measurement-based model. The cluster state is a *bipartite graph* and hence by König's theorem [35], its edge-chromatic number is $\Delta(G)$, hence from the above Lemma we conclude that any unitary can be implemented with a cluster state that can be prepared in depth 4. This however, might force the use of extra auxiliary qubits.

3.2. Quantum depth

The *quantum computation* depth of a pattern, or just *quantum* depth for short is the depth in the execution of the pattern that is due to the dependencies of measurement and correction commands on previous measurement results (this is also called the *causality depth*). Given a pattern in standard form, it is easy to calculate its quantum computation depth from its *execution* digraph given below.

Definition 3.2. The *execution digraph* R for a pattern $\mathcal{P}$ in standard form has V as node-set. Let the *domain* of a signal be the set of qubits on which it depends. The arcs of R are constructed in the following way:

1. Draw an arc from i to j whenever ${}_t[M_j]^s$ appears in the pattern, with i in the domain of s or t.
2. Draw an arc from i to j whenever X_j^s or Z_j^s appears in the pattern, with i in the domain of s.

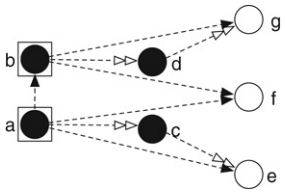

Fig. 2. The execution digraph for the standard pattern of Equation (18). A white double arrowed arc represents an X-dependency, and a black arrowed arc a Z-dependency. The X-dependency arcs correspond to edges of the underlying geometry, however this is not the case for the Z-dependency arcs.

We refer to the nodes of *in-degree* zero in R as *start* nodes. Similarly, the nodes of *out-degree* zero in R are called *end* nodes. If there is an arc from i to j in the execution digraph, we say that j *depends* (or has a *dependency*) on i. As a consequence of the definiteness condition the graph of any pattern is acyclic and hence we can give the following definition for the quantum computation depth:

Definition 3.3. Let $\mathcal{P}$ be a pattern in standard form. The *quantum computation depth* for $\mathcal{P}$ is the number of vertices on the longest directed path between a start and end node in the execution digraph. We call such a longest path a *critical path*.

It is trivial that for standard patterns with flow, the quantum computation depth is the same as the flow depth, as flow defines the dependency structures of the pattern.

As an example, consider again the geometry given in Fig. 1, one can write a uniformly and strongly deterministic pattern on this geometry using the flow theorem that can be rewritten in the following standard form:

$$Z_g^{s_b} X_g^{s_d} Z_f^{s_b} Z_f^{s_a} Z_e^{s_a} X_e^{s_c} [M_d^{\delta}]^{s_b} [M_c^{\gamma}]^{s_a}{}_{s_a} [M_b^{\beta}] M_a^{\alpha} E_G, \tag{18}$$

where G is the entanglement graph corresponding to the geometry of Fig. 1. Following Definition 3.2, the execution digraph for the above pattern is given in Fig. 2. As said before, there are two types of dependent measurements defined by X and Z-dependencies (Eqs. (2) and (3)), that are represented with different arrows in Fig. 2. The longest path in the execution digraph is $abdg$, hence from Definition 3.3, the pattern depth is 4.

Recall that any wild pattern consists of a sequence of standard sub-patterns where the E commands are interspersed within the pattern. Naturally, we define the depth of a wild pattern to be the sum of the depths (preparations and execution) of its standard parts. In order to define this combined preparation and quantum depth, we define the execution digraph in a similar way as Definition 3.2, but we add the E commands to the execution digraph. Then the *depth* of a wild pattern is the longest path in the execution digraph *except* that we allow a sequence of E commands to be parallelized, the depth of such a sequence being given by the results of Section 3.1.

3.3. Classical depth

The classical computation of the correction exponents as well as the measurement angles introduces the *classical* depth of a pattern. Consider, for example, the case where we have a correction of the form $X_i^{s_1+s_2+\cdots+s_n}$. An efficient implementation would start by classically calculating the parity of the exponent, and then applying the correction if the parity is 1. This is also the case for a measurement angle such as $[M_i^{\alpha}]^{s_1+s_2+\cdots+s_n}$, where one needs to delay the quantum computation to classically compute the measurement angle. Luckily all these classical delays are of at most $O(\log(n))$ depth, since the parity of n bits can be computed by a divide-and-conquer method in depth $O(\log(n))$ (any polynomial-size parity circuit has depth in $\Omega(\log^* n)$ [36]). Such a classical computation cost between quantum layers is negligible, but it still exists. Actually, depending on the underlying geometry of a pattern, this classical processing sometimes requires only constant depth. This can be easily seen for a pattern with flow.

Lemma 3.4. *Let $\mathcal{P}$ be a standard pattern with flow and geometry G, before signal shifting has been performed. The depth of the classical processing required between quantum layers is in $O(\log \Delta(G))$.*

Proof. It is enough to show that the number of terms in the dependency of measurement at qubit i is at most equal to the degree of vertex i in G. This is evident from the flow theorem and Eq. (17) as there is at most one term in the X-dependency and the rest of the neighbors of i contribute at most one Z-dependency. Hence the depth for the classical computation required for calculating the measurement angles at qubit i is in $O(\log(\deg(i)))$. Therefore, at each qubit, the classical depth is in $O(\log \Delta(G))$. □

Therefore, for a simple geometry such as the cluster state with maximum degree 4, all classical computation is constant. In the next section we present several methods for decreasing the quantum depth which might lead to the increase of classical depth. Note that the quantum versus classical depth tradeoff (Proposition 4.4) is beneficial as classical depth will not contribute towards the decoherence of the underlying quantum system.

4. Quantum depth-reducing techniques for patterns

In this section, we refer to the combined preparation and quantum depth of a standard pattern as its *depth* and we present our main automated techniques for reducing the depth of patterns. While the results of Sections 4.1 and 4.2 deal with the depth of *any* pattern, we will also present better bounds for the depth of patterns with *flow*. The class of patterns with flow is an interesting class of patterns, as it is universal for quantum computing, closed under composition and more importantly our translation from circuits to patterns in Section 5 always yields a pattern with flow. We will use the following important notion of *influencing walks* for geometries.

Definition 4.1. Let $(f, \preceq)$ be the flow of a geometry (G, I, O). Any input–output walk in G that starts with a flow edge, has no two consecutive non-flow edges and traverses flow edges in the forward direction, is called an *influencing walk*.

Similarly a *partial influencing walk* from node v is an I–v walk in G that starts with a flow edge at an input node in I, ends with a flow edge at node v, contains no consecutive non-flow edges and traverses flow edges in the forward direction. The following are examples of several influencing walks in the geometry with flow of Fig. 1:

ace, acf, $acbdf$, $acbdg$.

4.1. Standardization

Intuitively, we would expect that standardization could only potentially decrease the depth. This is because by standardizing, we benefit from the fact that there is a single entanglement graph to consider. Also, corrections are propagated to the end and applied only on output qubits, hence potentially fewer operations are needed. On the other hand, standardization creates dependent measurements. The following theorem (which is general and independent of the flow construction) confirms all these observations. Let $\mathcal{P} \Rightarrow^{\star} \mathcal{P}'$ denote the fact that $\mathcal{P}'$ is obtained from $\mathcal{P}$ by applying a finite sequence of rewrite rules given by Eqs. (6)–(9).

Theorem 4.2. *Whenever $\mathcal{P} \Rightarrow^{\star} \mathcal{P}'$ where P' is in standard form, the depth of $\mathcal{P}'$ is less than or equal to the depth of $\mathcal{P}$.*

Proof. Since the depth of $\mathcal{P}$ is the sum of the depths of its standard parts, it is sufficient to show that standardization of a wild pattern $\mathcal{P}$, containing two parts in standard form, say $\mathcal{P} = C^2M^2E^2C^1M^1E^1$ (where some or all of the parts may be empty), does not increase its depth. The theorem then follows by induction.

Step 1. (The E's)

We show how the re-writing rules are used to bring the pattern $\mathcal{P} = C^2M^2E^2C^1M^1E^1$ to $\mathcal{P}' = C^2M^2C^{1'}M^1E^2E^1$ and that by doing so, the depth of $\mathcal{P}'$ is no greater than that of $\mathcal{P}$. The result holds trivially, if E^2 is empty. Otherwise, for every command $E_{ij} \in E^2$, commute it to the right-hand side of the pattern by doing the following:

1. If C^1 contains Z_i or Z_j corrections, but no X_i or X_j corrections, we apply the rewriting rule $E_{ij}Z_i^s \Rightarrow Z_i^sE_{ij}$ and hence the depth does not increase. We then complete the commutation by applying the free commutation rules.
2. If C^1 contains X_i or X_j, then the rewriting rule $E_{ij}X_i^s \Rightarrow X_i^sZ_j^sE_{ij}$ applies. Here, the command X_i^s has an s dependency, which obviously cannot contain i or j, since these qubits have not been measured yet. Since X_i^s and Z_i^s do not depend on each other, the addition of the extra correction does not contribute to the depth.We then complete the commutation by applying the free commutation rules.

Finally, consider the entanglement graph for E^1E^2. Since $\chi'(E^1 \cup E^2) \leq \chi'(E^1) + \chi'(E^2)$, clearly, the preparation depth for $E^1 \cup E^1$ cannot be any greater than the depth of preparation for E^1 plus E^2. Also, as an extra bonus, since E_{ij} is self-inverse, if E^1 and E^2 happen to have common commands, they will cancel out.

Step 2. (The M's)

We will show how the free commutation rules and the re-writing rules are used to bring the pattern $\mathcal{P}' = C^2M^2C^{1'}M^1E^2E^1$ to its standard form $\mathcal{P}'' = C^2C^{1''}M^{2'}M^1E^2E^1$ and that doing so, the depth of $\mathcal{P}''$ is no greater than that of $\mathcal{P}'$.

1. Consider a command ${}_t[M_i^{\alpha}]^s \in M^2$. If $C^{1'}$ does not contain any commands acting on qubit i, then ${}_t[M_i^{\alpha}]^s$ freely commutes in $C^{1'}$, hence we have commuted ${}_t[M_i^{\alpha}]^s$ to the right-hand side of $C^{1'}$, and clearly the sum of the depths of the patterns C^2M^2 and $C^{1'}M^1E^2E^1$ is greater than or equal to the depth of the pattern $C^2C^{1'}M^2M^1E^2E^1$.
2. Otherwise, we apply the rewrite rules of Eqs. (8) and (9). This can only decrease the depth. □

Theorem 4.2 shows us that in order to improve the parallel run-time of the pattern, we should implement the standard form of the pattern. The following example demonstrates this property.

Example 1. Consider the following wild pattern with n input qubits with indices $\{1, \ldots, n\}$ and n output qubits with indices $\{n+1, \ldots, 2n\}$:

$$X_{2n}^{s_n}M_n^{\alpha_n}E_{n2n}E_{n(2n-1)} \cdots X_{i+n}^{s_i}M_i^{\alpha_i}E_{i(i+n)}E_{i(i-1+n)} \cdots X_{1+n}^{s_1}M_1^{\alpha_1}E_{1(1+n)}. \tag{19}$$

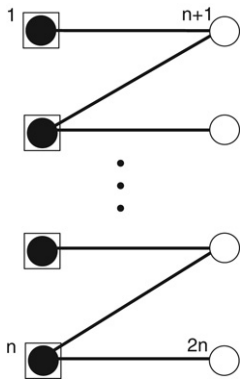

Fig. 3. A simple geometry with n inputs and n outputs to demonstrate the power of standardization and signal shifting in reducing depth asymptotically (see Examples 1 and 2).

The depth of the above wild pattern is $4n$ as it contains n standard sub-patterns with depth 4. After the standardization procedure, we obtain the following pattern, where G is the geometry in Fig. 3:

$$X_{2n}^{s_n} \cdots X_{i+n}^{s_i} \ \cdots \ X_{1+n}^{s_1} \ {}_{s_{n-1}}[M_n^{\alpha_n}] \ \cdots \ {}_{s_{i-1}}[M_i^{\alpha_i}] \ \cdots \ M_1^{\alpha_1} \ E_G. \tag{20}$$

The preparation depth for G is only 2 and the longest dependency path in the standard form is n. Therefore the depth of the pattern has decreased from $4n$ to n.

We also know that standardization can be performed in polynomial time [19]. Thus in the remainder of the paper, we will only consider standard patterns, which also allows us to consider the preparation depth separately from the quantum computation depth.

4.2. Signal shifting

The signal shifting rules (Eqs. (13)–(16)) tell us how we can push the Z dependencies of a pattern all the way to the end. The following theorem states that, in general, signal shifting does not increase the depth of a standard pattern. We then demonstrate with an example that it can indeed decrease the depth even exponentially.

Theorem 4.3. *Signal shifting for a standard pattern does not increase the depth.*

Proof. Let $\mathcal{P}$ be a pattern in standard form and suppose that $\mathcal{P}$ includes a command ${}_t[M_i^\alpha]^s$ which generates the signal shifting command S_i^t. Let $\mathcal{P}'$ be the pattern that corresponds to the pattern after the signal S_i^t has been shifted. Let D be the execution digraph for $\mathcal{P}$ and let D' be the execution digraph for $\mathcal{P}'$. We want to show that the length of a critical path of D' is no greater than the length of a critical path of D.

Suppose that the domain of s in ${}_t[M_i^\alpha]^s$ is $s_1, s_2, \ldots, s_n$ and that the domain of t is $t_1, t_2, \ldots, t_m$. Consider all the commands that appear *after* ${}_t[M_i^\alpha]^s$ in $\mathcal{P}$ and that have an i dependency; denote these commands $C_{a_1}^i, C_{a_2}^i, \ldots, C_{a_k}^i$ (these are either corrections or measurements). We will show that the depth does not increase when we shift the signal S_i^t passed all the C_a^i's.

Consider the arcs in D that represent the dependencies between the measurement ${}_t[M_i^\alpha]^s$ and measurements of qubits $t_1, t_2, \ldots, t_m$; these are the arcs $t_j i$ (for $j = 1 \ldots m$) and we will call these the *old* arcs. So the old arcs represent Z-dependencies for the ${}_t[M_i^\alpha]^s$ measurement. These are precisely those that create signal shifting commands, since ${}_t[M_i^\alpha]^s \Rightarrow S_i^t\,[M_i^\alpha]^s$.

Now consider the arcs in D' that represent the dependencies between the measurement of qubit t_j, M_{t_j} ($j = 1 \ldots m$) and the measurements and corrections that have an i dependency, $C_{a_x}^i$ ($x = 1 \ldots k$). We call these arcs *new* arcs since they represent the new dependencies created by $S_i^{t_1}, S_i^{t_2}, \ldots, S_i^{t_n}$ by the signal shifting rules given in Eqs. (13)–(16).

Indeed, when we apply signal shifting to $\mathcal{P}$, we get rid of all the dependencies represented by old arcs, yet we add all the dependencies represented by new arcs. These are the only differences between the execution digraphs D and D'.

If all new arcs are already in D (this could be the case if all the dependencies were present before signal shifting), the graph D' cannot have a longer critical path than D and we are done. Otherwise, suppose for a contradiction that the length of a critical path in D' is greater than the length of a critical path in D. Since D' differs from D only by the removal of all the *old* arcs and the addition of all the *new* arcs, the only way for D' to have a longer critical path than D would be for this critical path to include a *new* arc, say $t_j a_k$ (obviously, the removal of the *old* arcs in D' cannot contribute to a longer critical path). But if such a critical path exists in D', then D admits a longer critical path, namely the same critical path in D', but with arcs $t_j i$ and $i a_k$ instead of arc $t_j a_k$. This contradiction proves our claim. □

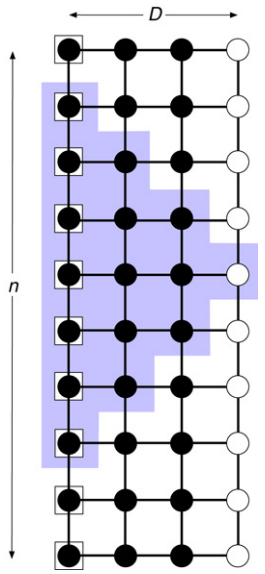

Fig. 4. A full cluster state geometry where the pyramid shape presents the backward cone, the set of all influencing walks that lead to a qubit. Only vertices on the pyramid contribute to the classical depth complexity of the command to be performed at that qubit.

There is a tradeoff when we perform signal shifting, as it can increase the classical depth. However, as we show below, the classical depth is at most $O(\log(n))$ at each layer, where n is the number of measured qubits and hence the tradeoff is beneficial, especially from the point of view that classical computation is cheap and reliable, compared to quantum computation that is expensive, error-prone and subject to decoherence.

Proposition 4.4. *Let $\mathcal{P}$ be a pattern with flow where standardization and signal shifting have been performed. Fix a node v in the underlying geometry G and let I_v be the set of all partial influencing walks, from an input qubit i to the node v. (If v is an output qubit we consider all the influencing walks instead.) Let N_v be the set of vertices that are on any walk in I_v. Then the classical depth of the required classical computation for computing the angles of measurement command or the exponent of correction command at v is in $O(\log |N_v|)$.*

Proof. In the proof of Lemma 3.4, we saw how the flow theorem tells us which dependencies are applied to a qubit v. Once signal shifting has been done, the dependencies are modified, but they still propagate only through influencing walks. In fact for the case of v being a measured qubit, after signal shifting only the X-dependencies remain and therefore we need to consider only the partial influencing walks. Hence there are at most $|N_v|$ dependencies at v; the parity of these dependencies can be computed in classical depth $O(\log |N_v|)$. □

Note that N_v is upper bounded by the total number of qubits in the pattern. However for a particular geometry and angles of measurement, it can be smaller. For example, consider a pattern with n input qubits and a geometry of a full cluster state of size n times width equal to D, as shown in Fig. 4. Then from the above proposition, we conclude that the classical depth is $O(\log(D)$: for any given qubit i only the $O(D^2)$ qubits siting on the pyramid with qubit i as the top of the pyramid, will contribute to the depth complexity of the command to be performed at qubit i (see Fig. 4). Therefore, for $D \in O(\log(n))$, we obtain small classical depth of size $O(\log(\log(n)))$, whereas the total number of the qubits in the pattern is in $O(n \log(n))$.

We conclude with the continuation of Example 1 to demonstrate the power of signal shifting in reducing the depth asymptotically.

Example 2. Consider the standard pattern of Example 1 with depth n:

$$X_{2n}^{s_n} \cdots X_{i+n}^{s_i} \cdots X_{1+n}^{s_1} \; {}^{s_{n-1}}[M_n^{\alpha_n}] \cdots {}^{s_{i-1}}[M_i^{\alpha_i}] \cdots M_1^{\alpha_1} \; E_G. \tag{21}$$

After performing the signal shifting procedure we obtain:

$$X_{2n}^{s_1+\cdots+s_n} \cdots X_{n+1}^{s_1} \; M_n^{\alpha_n} \cdots M_1^{\alpha_1} \; E_G. \tag{22}$$

All the measurements can be performed at the same time and hence the pattern has quantum depth $2+2$, but the classical depth has now increased to $\log(n)$. This example demonstrates an exponential advantage that one can obtain via signal shifting while increasing the classical depth only logarithmically.

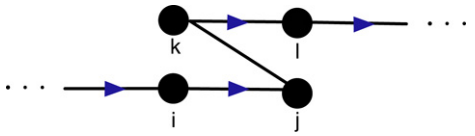

Fig. 5. Part of an influencing walk where two sequences of consecutive flow edges are connected with a non-flow edge.

4.3. Flow and influencing walk

It is known that for patterns with flow and equal input and output number of qubits, i.e. those implementing a unitary operator, the flow, if it exists, has a unique successor function, f [32]. From this, we obtain an upper bound on the quantum computation depth directly from the underlying geometry by analyzing the structure of influencing walks.

Lemma 4.5. *In order to compute the quantum depth of a standard pattern with flow, it suffices to consider the depth along influencing walks.*

Proof. Let a and b be two qubits in a standard pattern with flow. We prove that if b depends on a, then a appears before b on a common influencing walk, and this holds both before and after signal shifting. This is a consequence of the flow theorem. Recall that before signal shifting, a measurement at a qubit j is X-dependent on the result of a measurement at another qubit i if and only if $j = f(i)$ that is, a flow edge between qubits i and j. Also a measurement at a qubit k is Z-dependent on the result of a measurement at another qubit i if and only if $j = f(i)$ and k is connected to j, that is a non-flow edge between qubits j and k connected to a flow edge between qubits i and j. Therefore signal shifting creates new dependencies only through influencing walks. Hence if qubit b depends on qubit a, it is either via a direct X or Z dependency or due to a sequence of dependencies after signal shifting, in all the cases a and b must be on a common influencing walk. □

It is easy to see that if a geometry has a flow, all of its influencing walks are of finite length. Note that after signal shifting, Z-dependencies coming from the non-flow edges on an influencing walk no longer contribute to the pattern depth, as the dependencies that they represent are pushed to the final correction on an output qubit. On the other hand, signal shifting can create new X-dependencies. The next proposition presents an upper bound on the effect of signal shifting on the pattern depth. It also takes into account the effect of the Pauli measurements on depth due to the following identities.

$$M_i^{\frac{\pi}{2}} X_i^s = M_i^{\frac{\pi}{2}} Z_i^s \tag{23}$$

$$M_i^0 X_i^s = M_i^0. \tag{24}$$

According to Eq. (23), when a qubit i is measured with angle $\frac{\pi}{2}$ (Pauli Y measurement), then any X-dependency on this qubit is the same as a Z-dependency. But after signal shifting, this Z-dependency does not directly contribute to the depth and hence we might obtain a smaller depth. Furthermore, there exists a special case where if qubit i is not an input qubit and also not the flow image of any other vertex ($\forall j : i \neq f(j)$) and qubit i is measured with $\frac{\pi}{2}$, then one can permit in the flow theorem, to have $f(i) = i$ and hence we will have one less flow edge [21]. This allows an influencing walk to have a loop edge on this particular vertex measured with Pauli Y and hence the influencing walk will not start with an input qubit. In the rest of the paper, we consider only this extended notion of influencing walk that takes into account the angles of measurement. When we want to emphasize this extended definition, we will refer to *Pauli influencing walks*.

According to Eq. (24), another special case is when qubit i is measured with angle 0 (Pauli X measurement), then any X-correction on qubit i can be ignored and in fact qubit i can be put at the first level of measurement. Consequently, again the flow depth can become smaller. By adding Eqs. (23) and (24) to the flow theorem, the proof still works [21] and we get a potential improvement on the depth complexity. We refer to this procedure as *Pauli simplification*. Another way of realizing these special cases is that after signal shifting, the Pauli measurements become independent measurements and hence can all be performed at the first level of the partial order. Hence in computing the depth of a pattern with flow after signal shifting is performed, one should disregard the Pauli measurements:

Proposition 4.6. *Let $\mathcal{P}$ be a pattern with flow where standardization, Pauli simplification and signal shifting have been performed. Let I_i be a Pauli influencing walk of $\mathcal{P}$, denote by e_i the number of the flow edges, by n_i the number of non-flow edges, by p_i number of flow edges pointing to a qubit to be measured with a Pauli measurement and by ℓ_i the number of loop edges ($\ell_i \in \{0, 1\}$). Then the depth of the pattern, call it $D_{\mathcal{P}}$ satisfies the following formula:*

$$D_{\mathcal{P}} \leq \max_{I_i} e_i - (n_i + p_i + \ell_i) + 1.$$

Proof. Consider an influencing walk I. The flow edges represent X-dependencies hence each flow edge in a sequence of consecutive flow edges contributes to the depth along I. Now, consider a configuration with a non-flow edge as shown in Fig. 5. Before signal shifting, the dependent measurements on qubits i, j, k and ℓ are given as follows (see Eq. (17)) where A, B and C stand for general signals not including s_i, s_j, s_k and s_ℓ

$$\ldots {}_D[M_\ell^{\alpha_\ell}]^{s_k}{}_{C+s_i}[M_k^{\alpha_k}]^B{}_A[M_j^{\alpha_j}]^{s_i} \ldots$$

and after signal shifting we have

$$\begin{aligned}
&\cdots {}_D[M_\ell^{\alpha_\ell}]^{s_k}{}_{C+\boxed{s_i}}[M_k^{\alpha_k}]^B{}_A[M_j^{\alpha_j}]^{s_i}\cdots\\
\Rightarrow\ &\cdots \boxed{{}_D[M_\ell^{\alpha_\ell}]^{s_k}S_k^{s_i}}{}_C[M_k^{\alpha_k}]^B{}_A[M_j^{\alpha_j}]^{s_i}\cdots\\
\Rightarrow\ &\cdots S_k^{s_i}{}_D[M_\ell^{\alpha_\ell}]^{s_k+s_i}{}_C[M_k^{\alpha_k}]^B{}_A[M_j^{\alpha_j}]^{s_i}\cdots
\end{aligned}$$

Therefore qubits j and ℓ are in the same layer. In other words, after signal shifting, the first flow edge after every non-flow edge does not contribute to the depth of the pattern. Also, any new X-dependency created with signal shifting will not increase the depth. Hence from the total number of flow edges on an influencing walk, we need to subtract the number of non-flow edges.

Now we consider the effect of the Pauli angles. Along any Pauli influencing walk, any flow edge pointing to a qubit to be measured by a Pauli X will not require a separate layer (Eq. (24)) and for the Pauli Y case, such a flow edge is converted to a Z-dependency (Eq. (23)), to be signal shifted as described above. Also if the influencing walk starts with a Y measurement followed by a non-Pauli measurement, we have a loop edge and hence the immediate following non-Pauli measurement can also be put in the first layer and hence we subtract the loop edge from the total depth for this influencing walk. Finally, due to Lemma 4.5 we take the maximum over all influencing walks and we have to add 1 to the depth since the depth is the number of vertices of influencing walk, and not the number of edges. □

4.4. Pauli resetting

We saw that the main ingredients to obtain a reduced pattern depth are influencing walks, Pauli measurements and signal shifting. In fact, Pauli measurements not only can be performed in the first layer but also can "reset" the pattern depth along an influencing walk. This intuition is formalized in the following theorem on pattern depth characterization. In what follows, we deal with sequences of measurement angles, where $N_1, N_2, \ldots$ represent non-Pauli measurements, X a Pauli X measurement, Y a Pauli Y measurement and P is either X or Y. Note that we use the same notation for a Pauli measurement angle and the Pauli measurement itself. Furthermore $(\omega)^*$ and $(\omega)^{\text{odd}}$ represents respectively, a non-negative and odd number of repetitions of ω.

The key to our result is the following Pauli simplification rule on the angles of measurement along an influencing walk.

Definition 4.7. Let I be an influencing walk, the simplified sequence of the angles are defined with the application of the following rule.

$$N\ \mathsf{P}_1\ \alpha_1\beta_1\ \mathsf{P}_2\ \alpha_2\beta_2\ \cdots\ \mathsf{P}_k\ N \Rightarrow \begin{cases} N & \text{if } \exists \mathsf{P}_i = (X)^{\text{odd}}(Y(X)^{\text{odd}})^* \\ N\,N & \text{otherwise} \end{cases}$$

where P_i represents a (possibly empty) finite sequence of Pauli measurements, and where a vertex that is incident to a non-flow edge along I either has its measurement angle recorded as α_i (if it is the tail of a flow edge), or as β_i (if it is the head of a flow edge).

Next we prove how the simplified sequence of angles characterizes the depth of an influencing walk and hence the depth of a pattern:

Theorem 4.8. *Let $\mathcal{P}$ be a standard pattern. The quantum depth of an influencing walk is $d+2$ if, after the above simplification, we obtain $\mathsf{P}\ N^d\ \mathsf{P}$ and is $d+1$ if we obtain either $Y\ N^d\ \mathsf{P}$ or $N^d\ \mathsf{P}$. The depth of $\mathcal{P}$ after Pauli simplification and signal shifting is given by the maximum depth over all influencing walks of $\mathcal{P}$.*

Proof. The first step is to show the unique property of the magical sequence

$$(X)^{\text{odd}}(Y(X)^{\text{odd}})^*$$

which resets the X-dependency between two non-Pauli measurements. More precisely, let i and j be two vertices on a common influencing walk I that are measured with non-Pauli angles and separated along I with only flow edges and the above Pauli sequence. Then after signal shifting, there will be no X-dependency between i and j.

Assume such an X-dependency between i and j exists, then it is necessarily due to the fact that during signal shifting, the last Pauli X measurement in the sequence acquires a Z-dependency from i; this Z-dependency would then be signal shifted to an X-dependency between i and j, since j has an X-dependency on the last Pauli X measurement. We use a parity argument to show that this never occurs.

Note that the sequence of Pauli measurements, $(X)^{\text{odd}}(Y(X)^{\text{odd}})^*$ is odd and through signal shifting, the Z-dependency that originates from i is shifted only through every even position in the Pauli measurement sequence. Due to the placement of the Y measurements which never occurs in an odd position, the special case of the Pauli Y rule (Equation (23)) cannot be applied to change the parity. Hence, the final X measurement in the sequence (which is at an odd position) never sees a Z-dependency from i (Fig. 6).

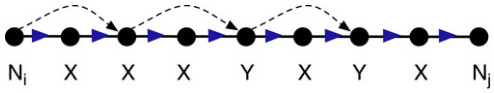

Fig. 6. Two non-Pauli measurements separated with the sequence of Pauli measurements of the form $(X)^{\text{odd}}(Y(X)^{\text{odd}})^*$. There is no Z-dependency between the last Pauli X measurement and the first non-Pauli measurement and therefore, after signal shifting, there will be no X-dependency between the non-Pauli measurements.

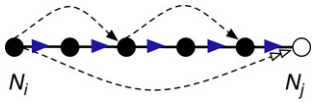

Fig. 7. An even number of Pauli measurements between two non-Pauli measurement leads to an X-dependency after signal shifting.

The next step is to show that Definition 4.7 defines the dependency structure along an influencing walk. Let i and j be two vertices on a common influencing walk I that are measured with non-Pauli angles N_i and N_j and separated along I with the following sequence of measurements angles:

$$\mathsf{P}_1\ \alpha_1\beta_1\ \mathsf{P}_2\ \alpha_2\beta_2\ \cdots\ \mathsf{P}_k$$

where each P_i is a (possibly empty) finite sequence of Pauli measurements along flow edges, and where a vertex that is incident to a non-flow edge along I either has its measurement angle recorded as α_i (if it is the tail of a flow edge), or as β_i (if it is the head of a flow edge).

We show that after signal shifting, there will be no X-dependency due to I between i and j if and only if at least one of the P_i sequence is equal to $(X)^{\text{odd}}(Y(X)^{\text{odd}})^*$. This will be also true even if one of i or j is an endpoint of a non-flow edge.

Assume i and j are connected with only flow edges (we have the sequence $N_i\mathsf{P}N_j$) and consider the following possible cases for the sequence P:

(I) It consists of an even number of Pauli angles. Then there is an X-dependency between i and j.
(II) It consists of an odd number of Pauli angles with at least one Y at an odd position from left to right. Then there is an X-dependency between i and j.
(III) It consists of an odd number of Pauli angles with no Y at any odd position: $(X)^{\text{odd}}(Y(X)^{\text{odd}})^*$. Then there is *no* X-dependency between i and j.

Fig. 7 shows how in Case (I), one obtains an X-dependency after signal shifting between i and j. Case (II) is also similar, by Pauli simplification, the X-dependency at a Y measurement of an odd position is considered as an Z-dependency and hence we obtain the same scenario as Case (I). Finally, Case (III) is proved before the lemma.

Now consider the case where there exists a non-flow edge between the non-Pauli angles and neither i nor j is an endpoint of a non-flow edge:

$$N_i\ \mathsf{P}_1\ \alpha\ \beta\ \mathsf{P}_2\ N_j.$$

According to the flow theorem, there is a Z-dependency from the qubit that precedes the qubit assigned to α angle to the qubit with angle β. In order to have a sequence of Z dependencies between i and the vertex with angle β, P_1 must satisfy the conditions of cases (I) or (II) and then similar to the above argument, in order to obtain an X-dependency between i and j, P_2 must also satisfy the conditions of cases (I) or (II) and hence we obtain the statement of the Lemma. The same argument is valid if either of i or j is an endpoint of a non-flow edge.

To finish the proof note that after applying the simplification rule, we obtain a unique final sequence of the form $\mathsf{P}\ N^d\ \mathsf{P}$ on any influencing walk and hence the longest sequence of dependent non-Pauli measurements will have length d and since there is a first layer of Pauli measurements and one final layer of corrections, the depth along this walk will be $d+2$. However, if the final form is $Y\ N^d\ \mathsf{P}$, then there will be no dependency between the Pauli Y and the first non-Pauli N (Eq. (23)) and depth is $d+1$ which is also the case for the final form $N^d\ \mathsf{P}$.

Finally, according to Lemma 4.5 the pattern depth is the maximum number of the dependent non-Pauli measurements along all the influencing walks and hence it is enough to compute the maximum value of i over all influencing walks. □

5. Circuits and measurement patterns

Having built all the required tools, we can now turn our attention to the main focus of the paper on parallelizing quantum circuits. To this end we give a method to translate a quantum circuit to a pattern (Section 5.1) and vice-versa (Section 5.2), where standardization, signal shifting and Pauli simplifications on the obtained pattern leads to a more parallel circuit. We also present the exact tradeoff for the transformations. Furthermore, our construction allows us to see influencing walks directly in the quantum circuit so that the pattern depth characterization results can be directly applied to circuits (Section 5.3).

We fix the universal family of gates to be $\mathfrak{U} = \{\wedge Z, J(\alpha)\}$:

$$\wedge Z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad J(\alpha) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\alpha} \\ 1 & -e^{i\alpha} \end{pmatrix} \quad \text{(for all angles } \alpha\text{)}.$$

In [28,37] it was shown that this family is universal for the circuit model since every single-qubit unitary operator can be written in terms of $J(\alpha)$:

$$U = e^{i\alpha} J(0)J(\beta)J(\gamma)J(\delta).$$

In addition, they lead to simple generating patterns. To implement the gate $J(\alpha)$ on the qubit i, it would be sufficient to add an auxiliary qubit j and to entangle it with i and implement the following measurement commands:

$$J(\alpha) := X_j^{s_i} M_i^{-\alpha} E_{ij} N_j \tag{25}$$

which defines a measurement pattern with input qubit i and output qubit j. Trivially the pattern to implement $\wedge Z_{ij}$ gates on two qubits i and j has input/output qubits $\{i, j\}$ with one simple command

$$\wedge Z := E_{12}. \tag{26}$$

Hence the family of $J(\alpha)$ and $\wedge Z$ gates is a good choice for translation between circuits and patterns and any other universal family can be replaced by this one with constant overhead. In the rest of the paper, whenever the angle α is not important, we simply refer to a $J(\alpha)$ gate as a J gate.

5.1. From circuits to patterns

The original universality proof for MBQC already contained a method to translate a quantum circuit containing arbitrary 1-qubit rotations and control-not gates to a pattern [16]. Here, we give an alternative method for the translation of a given circuit to a standard pattern in the MBQC so as to enable the reduction of the quantum depth. We give the exact tradeoff in terms of the number of auxiliary qubits and depth.

Recall that $\wedge Z$ is self-inverse and symmetric, hence any circuit that contains consecutive $\wedge Z$ gates acting on the same qubits can be simplified. In what follows, we suppose that this simplification has been performed.

Definition 5.1. Let C be a circuit of $\wedge Z$ and J gates on n logical qubits. The corresponding standard pattern $\mathcal{P}$ is obtained by replacing each gate in C with its corresponding pattern given by equations Eqs. (25) and (26), and then performing standardization and signal shifting.

In the above definition the sets V, I and O of the pattern are obtained as follows. After translating each gate into the corresponding sub-patterns, their input and output sets are defined according to equations (25) and (26). Now the input set I of the main pattern consist of those input qubits in the sub-patterns that corresponds directly to the input of the circuit. The output set O consists of all the non-measured qubits and V is the set of all the qubits that are defined in the translation. The next definition gives a procedure to obtain the corresponding pattern.

To present the exact tradeoff for the above translation, in particular to prove that the quantum depth cannot increase, we construct directly the underlying geometry of a given circuit. Following the literature, we refer to the circuit qubits as *logical* qubits. Other qubits that are added during construction of the entanglement graph will be referred to as *auxiliary* qubits.

Definition 5.2. Let C be a circuit of $\wedge Z$ and J gates on n logical qubits. The *labeled entanglement graph* G_C is constructed as a layer that is initially built on top of the circuit C by the following steps (see also the example of Fig. 8). We start from left to write:

1. Replace each $\wedge Z$ gate on logical qubits i and j with a vertical edge between two vertices: one on the ith wire and one on the jth wire. Label both vertices *Input/Output*. Replace each J gate on a logical qubit i with an horizontal edge between two vertices on the ith wire, label the left vertex *Input* and the right vertex *Output*.
2. To connect the above components, on each wire, start from the left and contract consecutive non-adjacent vertices as follows (the *contraction* of vertices v_1 and v_2 of a graph G is obtained by replacing v_1 and v_2 by a single vertex v, which is adjacent to all the former neighbors of v_1 and v_2):
   - Two vertices labeled *Input/Output* are contracted as one vertex with *Input/Output* label;
   - A vertex labeled *Input/Output* and a vertex labeled *Input* are contracted as one vertex with *Input* label;
   - A vertex labeled *Output* and a vertex labeled *Input/Output* are contracted as one vertex with *Output* label;
   - Two vertices labeled *Output* and *Input* are contracted as one vertex with *auxiliary* label.

It is easy to verify the following proposition that justifies the above construction.

Proposition 5.3. *The graph G_C obtained from Definition 5.2 is the entanglement graph for the measurement pattern that is obtained from Definition 5.1. Furthermore, input–output paths of vertices sitting on the same wire define the flow of G_C.*

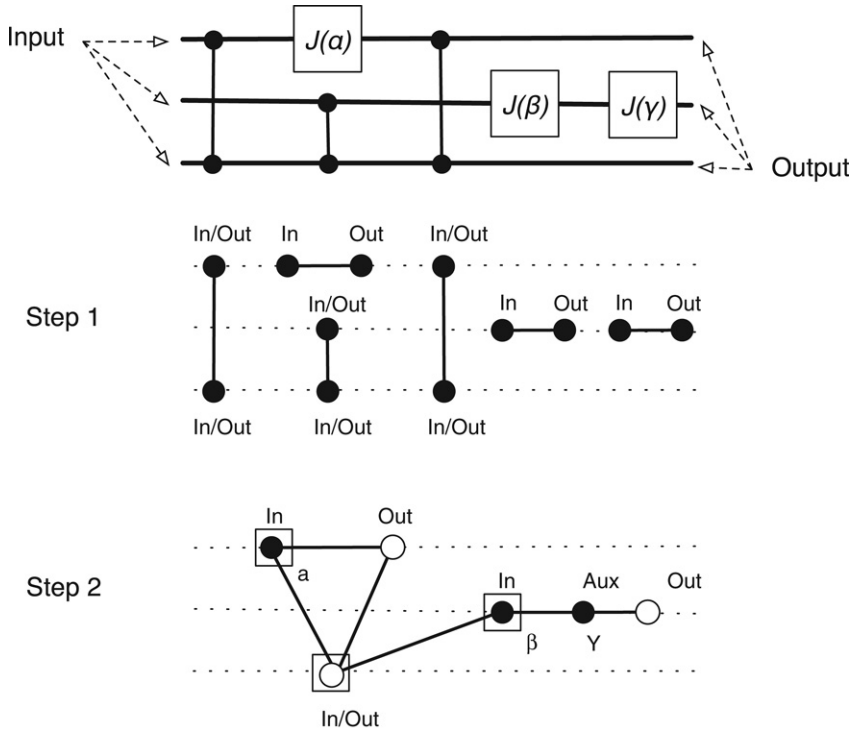

Fig. 8. A quantum circuit with $\wedge Z$ and $J(\alpha)$ gates, together with the two-step construction of the corresponding labeled entanglement graph. In the final step, an input qubit is represented by a boxed vertex and an output qubit with a white vertex. The black vertices will be measured with angles α, β and γ, as shown in the figure.

Proof. Recall that in Definition 5.1, we translate each gate to its sub-patterns and then apply the standardization procedure which does not change the underlying entanglement graph. Hence we will obtain a graph with input set being logical qubits, vertical edges corresponding to $\wedge Z$ gates and horizontal edges for J gates. This is the same construction obtained from Definition 5.2.

We know from the construction of Theorem 10 of [32] that for a geometry with $|I| = |O|$, a collection of vertex-disjoint $I - O$ paths in G_C defines the flow function f which is unique. Therefore, input–output paths of vertices sitting on the same wire define the flow of G_C. □

In order to obtain a full pattern corresponding to the circuit C, one needs to add measurement commands with angles being the same angles of the $J(\alpha)$ gates. These angles are assigned to the qubits labeled *Input* in Step (1) of the construction of Definition 5.2. The dependency structure is the one obtained from the flow theorem.

Proposition 5.4. *Let C be a quantum circuit on n logical qubits with only $\wedge Z$ and J gates. Let G_2 be the number of J gates and $D(n)$ the circuit depth. The corresponding pattern $\mathcal{P}$ given by Definition 5.1 has $n+G_2$ qubits, G_2 measurement commands, n corrections commands, and depth smaller than or equal to $D(n)$.*

Proof. The proof is based on the construction of Definition 5.2, which is obtained from replacing the patterns from Eqs. (25) and (26) for J and $\wedge Z$ gates and then performing the standardization procedure. It is clear from the construction that we start with n qubits corresponding to each wire, then any $\wedge Z$ connects the existing qubits (wires) and hence will not add to the total number of qubits. On the other hand any J gate extends the wire by adding a new qubit. This leads to the total number of $n + G_2$ qubits for the pattern. There are G_2 measurement commands since all but n qubits are measured. Since C has depth $D(n)$, any influencing walk in $\mathcal{P}$ has at most $D(n)$ flow edges. Hence the theorem is obtained from Proposition 4.6 after performing signal shifting on the corresponding pattern. □

Alternatively, for a given circuit, one can use another construction to obtain a corresponding pattern with cluster geometry, hence to achieve constant depth for the graph preparation stage. Naturally, the price is to have more qubits. First note that the following pattern implements teleportation from input qubit i to output qubit k that is simply the identity map (see Fig. 9):

$$X_k^{s_j} Z_k^{s_i} M_j^0 M_i^0 E_{jk} E_{ij}. \tag{27}$$

Now, if before Step (2) of the construction of Definition 5.2, we insert the teleportation pattern between any two consecutive $\wedge Z$ acting on a common wire, then the degree of each vertex remains less than 4 as desired. We will refer to this graph as the *cluster graph*, GC_C. In order to compute the number of qubits for the pattern obtained from this new

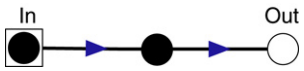

Fig. 9. The geometry of the teleportation pattern given in Eq. (27) with one input, one auxiliary and one output qubit.

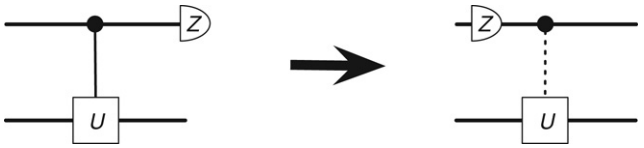

Fig. 10. A classically controlled implementation of a controlled-unitary gate. The computational basis measurement operator is represented by the half-circle box with Z label. After pushing the measurement to the beginning of the wire, the unitary U is only classically dependent (doted line) on the first wire.

construction, consider the positions in the circuit where two $\wedge Z$ appear after each other. These are the places where we need to apply the above teleportation pattern to keep the degree less than 4. With this construction, the depth of the pattern does not increase by more than a multiplicative constant. Therefore we have:

Lemma 5.5. *Let C be a quantum circuit on n qubits with only $\wedge Z$ and J gates. Let G_2 be the number of J gates, s the size of C and m the number of positions in C where two $\wedge Z$ appear after each other. Then the pattern $\mathcal{P}$ with the cluster graph construction (obtained as in Proposition 5.3 with the addition of the teleportation pattern above) has $n + G_2 + m \in O(n + s)$ qubits and depth in $O(D(n))$.*

In what follows, we always assume the cluster geometry for patterns corresponding to a circuit and hence the preparation depth is 4 (Section 3.1).

5.2. From patterns to circuits

The construction of Definition 5.2 can be also used in reverse order to transfer a pattern with flow to a corresponding circuit, where all the auxiliary qubits will be removed and hence by doing so the quantum depth might increase. However, we now show how to obtain another transformation from patterns to circuits where one keeps all the auxiliary qubits. This construction is simply based on the well-known method of coherently implementing a measurement. Recall that a controlled-unitary operator where the control qubit is measured in the computational basis $\{|0\rangle, |1\rangle\}$ can be written as a classical controlled unitary by pushing the measurement before the controlled-unitary operator [9], see Fig. 10.

Given a pattern in the standard form, we use the above scheme in the reverse order to convert the classically dependent measurements and corrections, and then push all the independent measurements to the end of the pattern. However since the scheme works only for the computational basis measurement, we have to first simplify all the arbitrary measurements M^α. Let $Z(\alpha)$ be the phase gate and H the Hadamard gate and let M^Z be the computational basis measurement (i.e. Pauli Z measurement). Then we have

$$M^\alpha = M^{\{|+_\alpha\rangle,|-_\alpha\rangle\}} = M^{HZ(-\alpha)^\dagger\{|0\rangle,|1\rangle\}} = M^Z HZ(-\alpha). \tag{28}$$

Additionally, we replace any classical X- and Z-dependencies of measurements and any dependent corrections with a sequence of $\wedge X$ and $\wedge Z$, which might create a quantum depth linear in the number of the dependencies, as shown in Fig. 11. However to reduce this linear depth, we can use the following result on parallelizing a circuit with only controlled-Pauli gates to logarithmic depth:

Proposition 5.6 ([11]). *Circuits on n qubits consisting of controlled-Pauli gates and the Hadamard gate can be parallelized with a circuit with $O(\log n)$ depth and $O(n^2)$ auxiliary qubits.*

We can now formalize the above translation of patterns to circuits.

Definition 5.7. Let $\mathcal{P}$ be a standard pattern with computational space (V, I, O), underlying geometry (G, I, O) (where G has a constant maximum degree) and command sequence (after signal shifting):

$$\cdots C_i^{S_i} \cdots [M_i^{\alpha_i}]^{S_i} \cdots E_G$$

where S_i is the set of qubits that the measurement or correction of qubit i depends on. Note that due to the signal shifting, we only have X dependencies. The corresponding *coherent circuit* C with $|I|$ logical qubits and $|V \setminus I|$ auxiliary qubits, is constructed in the following steps (see also Fig. 11):

1. Apply individual Hadamard gates on all the auxiliary qubits.
2. Apply a sequence of $\wedge Z$ gates according to the edges of G.
3. Replace any dependent measurement $[M_i^{\alpha_i}]^{S_i}$ with $M_i^Z H_i Z_i(-\alpha) \wedge_{S_i,i} X$ where $\wedge_{S_i,i} X$ is a sequence of controlled-X with control qubits in S_i and target qubit i. Note that since the M^Z is independent and can be pushed to the end of the corresponding wire it can be discarded.

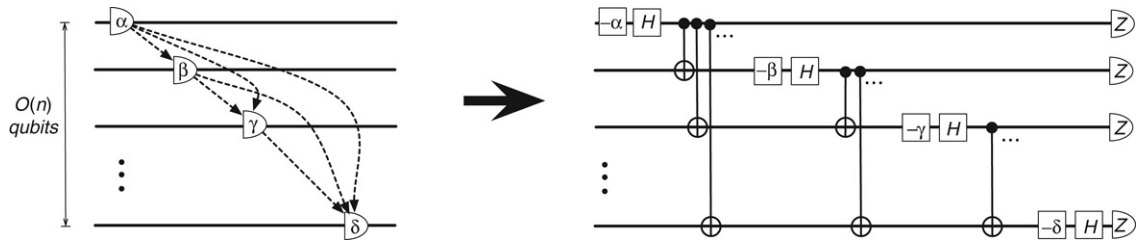

Fig. 11. Implementing coherently the sequence of dependent measurements in a pattern. An arbitrary measurement M^{α} is represented by a half circle labeled with its angle. The Hadamard and phase gates are shown with square boxes with the labels being H or the angle of the phase gate. The dotted arcs represent X-dependencies. Eq. (28) is used to simplify the measurements. After replacing the X-dependencies by $\wedge X$ gates, we obtain a quantum depth linear in the number of dependencies.

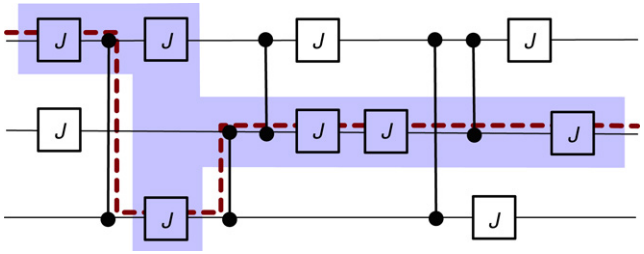

Fig. 12. A circuit with one of its influencing paths presented as a doted line. The J gates in the shaded area are those referred to as the J gates of the walk.

4. Replace any dependent correction $X_i^{S_i}$ with $\wedge_{S_i,i}X$ and $Z_i^{S_i}$ with $\wedge_{S_i,i}Z$.
5. Replace the joint sequence of added $\wedge X$ and $\wedge Z$ in steps 3 and 4 with the parallel form obtained from Proposition 5.6.

Lemma 5.8. *Let $\mathcal{P}$ be a standard pattern with computational space (V, I, O) and underlying geometry (G, I, O) (where G has a constant maximum degree). Let $t = |V \setminus O|$ be the number of measured qubits and let d be the quantum computation depth of $\mathcal{P}$. Then the corresponding coherent circuit C obtained from Definition 5.7 has $|I|$ logical qubits, $O(t^3)$ auxiliary qubits and depth $O(d \log t)$.*

Proof. We examine the cost at each step of the construction of Definition 5.7. Steps 1 and 2 add a constant to the depth of C. At step 3, each measurement has as most t dependencies, which, in step 5 translates to $O(\log t)$ depth with $O(t^2)$ auxiliary qubits. At step 4, each output qubit has at most t dependencies, which again in step 5 translates to $O(\log t)$ depth with $O(t^2)$ auxiliary qubits. Since the depth of $\mathcal{P}$ is d, the total depth of C is $O(d \log t)$, with $O(t^3)$ auxiliary qubits. □

Note that the logarithmic increase in the depth of C is due to the fact that the circuit model does not exploit any classical dependencies. Thus the classical computation of the measurement angles and corrections in $\mathcal{P}$ contributes to the quantum depth in C.

One can combine the forward and backward construction from circuit to patterns to obtain an automated rewriting system for the circuit which can decrease the depth by adding auxiliary qubits. The following theorem gives the tradeoff.

Theorem 5.9. *Let C be a quantum circuit on n qubits with only $\wedge Z$ and J gates. Suppose C has size s and depth D. Assume further that $\mathcal{P}$ is the corresponding pattern obtained from the forward translation as in Lemma 5.5 and that $\mathcal{P}$ has quantum depth D' (we know that $D' \leq D$). Then circuit C' constructed from $\mathcal{P}$ by Definition 5.7 has $O(s^3 + n)$ qubits, and depth in $O(D' \log s)$.*

Proof. The first step is to translate C to a pattern $\mathcal{P}$ using Lemma 5.5. The resulting pattern $\mathcal{P}$ has $O(s + n)$ qubits, and quantum depth $O(D)$. Then we translate the pattern back to a circuit C' using Definition 5.7. By Lemma 5.8, the new circuit has $O(s^3)$ auxiliary qubits and depth $O(D' \log s)$. □

At first glance it seems like applying Theorem 5.9 to a quantum circuit would not necessarily be beneficial, since the number of auxiliary qubits and the depth seem to increase. But note that we have given only upper bounds. As we showed in Section 4.3, taking into account Pauli simplification and signal shifting can give a significant improvement. In fact from pattern depth characterization result, Theorem 4.8, we present in the next section a characterization of those circuits to which applying Theorem 5.9 will necessarily decrease the depth.

5.3. Parallelizing circuits

In order to present the pattern depth characterization result directly in terms of the circuit language, we first define the notion of *circuit influencing path*.

Definition 5.10. Let C be a circuit of $\wedge Z$ and J gates. A left-to-right path starting at the beginning of a circuit wire and ending at any wire, such that the jumps between wires are done through $\wedge Z$ gates is called a *circuit influencing path* if there exist no two consecutive jumps i.e. without a J gate in between (see Fig. 12).

Recall that for patterns with an equal number of input and output qubits, the flow, if it exists, is unique. Hence it is easy to verify that circuit influencing paths defined above are exactly influencing walks of the corresponding pattern via the direct translation given in Section 5. Similar to the pattern case, the circuit depth is characterized in terms of the sequence of J gates appearing on the influencing paths defined below.

Definition 5.11. Let I be a circuit influencing path of circuit C. The set of J gates over I is defined to be all the consecutive J gates over the wires of the path including the J gates just after a $\wedge Z$ gate of a jump, as shown in Fig. 12.

Given an influencing path, we define the following simplification rule for the J gates:

$$J\ \mathsf{P}_1\ A_1B_1\ \mathsf{P}_2\ A_2B_2\ \cdots\ \mathsf{P}_k\ J \Rightarrow \begin{cases} J & \text{if } \exists \mathsf{P}_i = (H)^{\text{odd}}(H^i(H)^{\text{odd}})^* \\ JJ & \text{otherwise.} \end{cases}$$

where P_i represents a finite sequence of H and H^i gates and A_i and B_i represents the J gates immediately after a controlled-Z gate on the circuit influencing path (on the control and the target wires).

Theorem 5.12. *Let C be a circuit of gates in $\mathcal{U}$ on n qubits with size s and depth D. Assume that after the above simplification rule, over all circuit influencing paths, we obtain at most D' many consecutive J gates. Then circuit C can be parallelized to an equivalent circuit C' with depth in $O(D' \log(s))$ and size in $O(s^3 + n)$.*

Proof. Recall that $J(0) = H$ and $J(\frac{\pi}{2}) = H^i$ and hence the X and Y measurements in a pattern represent the H and H^i gates of the corresponding circuit, and the above circuit simplification rule translates into the same simplification rule for pattern i.e. Definition 4.7 and the proof is obtained from Theorem 4.8. Finally the bounds are direct result of Theorem 5.9. □

6. Applications

In this section we present several applications of our main results, Theorems 4.8 and 5.12.

Proposition 6.1. *Let $\mathcal{P}$ be a pattern with flow f, where standardization, Pauli simplification and signal shifting have been performed. The quantum computation depth is equal to 2 if and only if any qubit measured with a non-Pauli angle is not the flow image of any other vertex and hence it is either an input qubit or is connected to a vertex with a loop flow edge.*

Proof. Due to Theorem 4.8, $\mathcal{P}$ has depth 2 if and only if on all the influencing walks, after the simplification rule, we obtain one of the following final forms for the sequence of the measurement angles:

$$N\,\mathsf{P} \quad \text{or} \quad Y\,N\,\mathsf{P} \quad \text{or} \quad \mathsf{P}.$$

Now consider only those influencing walks with only flow edges, by reverse application of the simplification rules we conclude only input qubits can be measured with a non-Pauli angle or a non-input qubit measured by a non-Pauli measurement should not be the flow image of any other qubit and be connected to a qubit measured with Pauli Y. □

Note that this proposition extends the previously know results that patterns with only Pauli measurements have depth 2 [8,29]. The following example shows how one can use Theorem 5.12 to construct low depth circuits. The main tool is the gate sequence

$$R = (H)^{\text{odd}}(H^i(H)^{\text{odd}})^*, \tag{29}$$

which if it is inserted between two J gates over a circuit influencing path will make them appear in the same layer of the final parallelized circuit.

As an application, consider the quantum circuit in Fig. 13 with size $O(n^2)$ and depth $O(n)$. Theorem 5.12 tell us how to parallelize it to depth $O(\log(n))$, while adding $O(n^6)$ auxiliary qubits. First note that on any circuit influencing path, any two J gates are separated by an R gate (Eq. (29)) and hence after the simplification rule, we will have no two consecutive J gates. In other words, the parameter D' in Theorem 5.12 is equal to 1 which implies the depth of the parallelized circuit will be in $O(\log(n))$.

It is easy to extend the circuit of Fig. 13 and still apply Theorem 5.12 to parallelize it with a circuit with depth in $O(\mathbf{poly}(\log(n)))$. On each wire, replace $O(\log(n))$ many J_{ij} gates with the following sequence of gates:

$$J_1\mathsf{P}_1J_2\mathsf{P}_2\ldots J_k \quad \text{with } k \in O(\log(n))$$

where P_i is a sequence of H and H^i gates of polynomial length. Now the parameter D' of Theorem 5.12 is in $O(\log(n))$ and the parallel circuit will have depth in $O(\log^2(n))$.

This set of examples, although somewhat artificially constructed, demonstrates how one might use Theorem 5.12 to construct a parallel circuit for a given problem in hand. We finish this section with several other results on circuit parallelization.

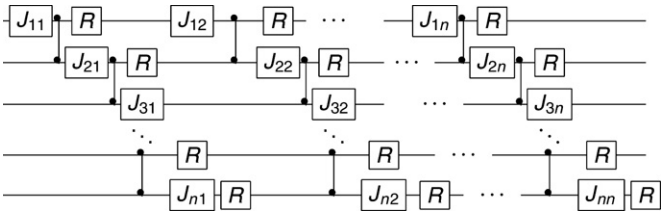

Fig. 13. A polynomial-depth circuit where each J_{ij} gate has an angle $\in [0, 2\pi)$ and the R gate stands for a sequence of Clifford gates of the form $(H)^{\text{odd}}(H^i(H)^{\text{odd}})^*$. Theorem 5.12 implies that this circuit can be parallelized with a logarithmic depth circuit.

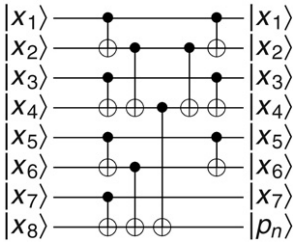

Fig. 14. A logarithmic-depth circuit for parity unitary transformation, where $p_n = \bigoplus_{i=1} x_i$.

Proposition 6.2. *A circuit on n qubits can be parallelized with a pattern of depth 2 via the construction given in Section 5 if and only if it is of the form: a possible sequence of individual phase gates, $Z_1(\alpha_1) \otimes \cdots \otimes Z_n(\alpha_n)$, followed by an arbitrary poly-size Clifford circuit.*

Proof. It is known that any Clifford gate can be implemented by a pattern with only Pauli X and Y measurements [19,20]. Hence in one direction, the proof is simply obtained by replacing the phase gates with qubits measured with a non-Pauli angles, that are input qubits. Then by Proposition 6.1, the corresponding pattern has depth 2.

To prove the other direction, let C be a circuit that can be parallelized with a pattern $\mathcal{P}$ with depth 2. Hence from Proposition 6.1 by adding appropriate $(Z(\alpha))^\dagger$ gates to the beginning of C, we obtain another circuit C' that translates to a pattern $\mathcal{P}'$ with only Pauli measurements. Now Theorem 4 in [19] implies that C' is in the Clifford group and hence C has the desired form. □

A simple case of the above proposition is for the case of a Clifford circuit, that was known already by [8,19,20,29]. On the other hand, the best known result in terms of depth complexity for the circuit implementing a subgroup of the Clifford group is Proposition 5.6 due to [11] which was then improved to the whole of Clifford group in [27]. Using our forward and backward construction of Section 5, we present another method for the parallelization of the whole Clifford group.

Proposition 6.3. *Any quantum circuit on n qubits of size $s \in$ **poly**(n) consisting of Clifford gates can be parallelized with a circuit with $O(\log n)$ depth and $O(s^3 + n)$ auxiliary qubits.*

Hence from Propositions 6.2 and 6.3, we see a logarithmic improvement in depth for implementations in the MBQC compared to the circuit model. What we achieve actually is a translation of quantum logarithmic depth in a circuit to constant quantum depth plus classical logarithmic depth in a pattern. We now show that this separation is tight by giving an example of a unitary that can be implemented as a pattern with constant quantum depth, but that *must* have logarithmic depth in the quantum circuit model.

Lemma 6.4. *Let U_p be the parity unitary transformation defined by*

$$U_p |x_1, x_2, \ldots, x_n\rangle = \left|x_1, x_2, \ldots, x_{n-1}, \bigoplus_{i=1}^{n} x_i\right\rangle.$$

Assume C to be any circuit consisting of 1- and 2-qubit gates that implements this unitary. Then the depth of C is in $\Omega(\log n)$.

Proof. Since the state of the last output qubit depends on every input qubit, and the circuit has only 1– and 2–qubit gates, the depth of the circuit must be in $\Omega(\log n)$. □

Fig. 14 gives a logarithmic depth circuit for U_p. This circuit uses only Clifford gates and hence by Proposition 6.2, we can implement it as a pattern with depth 2. Note however that the pattern has a classical logarithmic depth, which reconciles the depths in the two models: the *sum* of the classical and quantum depths in the pattern is equal to the total quantum depth in the circuit.

7. Discussions and future directions

The design of parallel algorithms is one of the main challenges in both classical and quantum computing and has a significant impact on theory and implementations. The advantage of quantum computing models over classical counterparts has been extensively studied in the context of computational complexity, whereas relatively little is known in terms of depth complexity. In addition, the comparison of models of quantum computing has been mainly explored from the computational aspect although other measures of comparison such as parallelism might lead to new directions in our understanding of the power and limitations of quantum computing.

In this paper, we considered two well-known models of quantum computing, the circuit model and the measurement-based model for quantum computing, and presented a logarithmic separation between them in terms of quantum depth complexity. We further demonstrated how a simple forward and backward transformation between circuits and measurement patterns leads to an automated procedure of parallelization. More importantly, the set of tools that we developed to study the depth complexity, such as the notion of the *influencing paths*, result in a simple construction for parallel patterns and circuits, this being the insertion of some particular type of Clifford operation among the non-Clifford ones.

A simple way of observing the advantages of the MBQC over the quantum circuit can be seen via the tradeoff between space and depth complexity as the transformation from a circuit to MBQC adds some auxiliary qubits and hence decreases the depth. On the other hand, one can also argue that the advantage is due to a clear separation of the types of depths that are involved in a computation: the preparation, quantum computation and classical depths. In other words, in the circuit model, all operations are done "*quantumly*" whereas in a pattern, some part of the computation can be performed via *classical processing*. This intuition seems to be also responsible for some of the previously known results on circuit parallelization such as the work of Robert Griffiths and Chi-Sheng Niu on the parallel semi-classical quantum Fourier transform [9]. Hence it would be interesting to see if our tools can indeed reproduce the same results for these or other classes of circuits where the output qubits are always measured.

Although it is encouraging that we obtain a generic method for circuit parallelization by exploiting the classical control structure in MBQC, it is not clear at this stage how our set of tools might be put in use to design parallel algorithms for a given classical problem and further work in this direction is necessary.

Another direction to investigate is the extension of the characterization results to the patterns with generalized flow [38], a recently developed notion for MBQC computing that provides both a necessary and sufficient condition for determinism[2] that might lead to a more parallel structure than patterns with flow.

Acknowledgements

We would like to thank Niel de Beaudrap for help in improving a previous version, Stephen Jordan for finding an error in our earlier upper bound for classical depth, and Iordanis Kerenidis for his insightful comments on circuit lower bounds techniques. We further thank Sean Barrett, Hugue Blier, Richard Cleve, Vincent Danos and Geňa Hahn for fruitful discussions. A. B.'s work is partially supported by scholarships from NSERC, FQRNT and the CFUW. E. K.'s work is partially supported by ARO-DTO. We are grateful to the CIFAR for financing E. K.'s stay at the Université de Montréal where this collaboration began.

References

[1] T. Pellizzari, Quantum networking with optical fibres, Physical Review Letters 79 (1997) 5242–5245.
[2] J.I. Cirac, P. Zoller, H.J. Kimble, H. Mabuchi, Quantum state transfer and entanglement distribution among distant nodes in a quantum network, Physical Review Letters 78 (1997) 3221–3224.
[3] S.C. Benjamin, Schemes for parallel quantum computation without local control of qubits, Physical Review A 61 (2000) 020301.
[4] E. Knill, R. Laflamme, G.J. Milburn, A scheme for efficient quantum computation with linear optics, Nature 409 (2001) 46–52.
[5] M.A. Nielsen, Optical quantum computation using cluster states, Physical Review Letters 93 (2004) 040503.
[6] D.E. Browne, T. Rudolph, Resource-efficient linear optical quantum computation, Physical Review Letters 95 (2005) 010501.
[7] S.D. Barrett, P. Kok, Efficient high-fidelity quantum computation using matter qubits and linear optics, Physical Review A 71 (2005) 060310(R).
[8] R. Jozsa, An introduction to measurement based quantum computation. Available as http://arxiv.org/abs/quant-ph/0508124v2, 2005.
[9] R.B. Griffiths, C. Niu, Semiclassical Fourier transform for quantum computation, Physical Review Letters 76 (1996) 3228–3231.
[10] R. Cleve, J. Watrous, Fast parallel circuits for the quantum Fourier transform, in: Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science (FOCS 2000), p. 526–536, 2000.
[11] C. Moore, M. Nilsson, Parallel quantum computation and quantum codes, SIAM Journal on Computing 31 (2002) 799–815.
[12] F. Green, S. Homer, C. Pollett, On the complexity of quantum ACC, in: Proceedings of the 15th Annual IEEE Conference on Computational Complexity, 2000, pp. 250–262.
[13] F. Green, S. Homer, C. Moore, C. Pollett, Counting, fanout and the complexity of quantum ACC, Quantum Information & Computation 2 (2002) 35–65.
[14] K. Iwama, S. Yamashita, Transformation rules for CNOT-based quantum circuits and their applications, New Generation Computing 21 (2003) 297–317.
[15] D. Maslov, G. W. Dueck, D. M. Miller, Quantum circuit simplification and level compaction, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 27 (3) (2008) 436–444. Available as http://arXiv.org/quant-ph/0604001, 2006.
[16] R. Raussendorf, H.J. Briegel, A one-way quantum computer, Physical Review Letters 86 (2001) 5188–5191.
[17] M.A. Nielsen, Cluster-state quantum computation, Reports on Mathematical Physics 57 (2006) 147–161.
[18] D.E. Browne, H.J. Briegel, One-way quantum computation — a tutorial introduction. Available as http://arxiv.org/abs/quant-ph/0603226v2, 2006.

[2] More precisely, for strong and stepwise determinism [38].

[19] V. Danos, E. Kashefi, P. Panangaden, The measurement calculus, Journal of the ACM 54 (2) (2007) 45 pp. Article No. 8.
Available as http://arxiv.org/abs/0704.1263v1.
[20] R. Raussendorf, H. Briegel, Computational model underlying the one-way quantum computer, Quantum Information & Computation 2 (2002) 443–486.
[21] V. Danos, E. Kashefi, Determinism in the one-way model, Physical Review A 74 (2006) 052310.
[22] R. Feynman, Simulating physics with computers, International Journal of Theoretical Physics 21 (1982) 467–488.
[23] D. Deutsch, Quantum theory, the Church–Turing principle and the universal quantum computer, Proceedings of the Royal Society of London A 400 (1985) 97–117.
[24] D. Deutsch, Quantum computational networks, Proceeding of the Royal Society of London A 425 (1989) 73–90.
[25] M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, Cambridge, 2000.
[26] D. Gottesman, Stabilizer codes and quantum error correction, PhD Thesis, California Institute of Technology, 1997.
[27] S. Aaronson, D. Gottesman, Improved simulation of stabilizer circuits, Physical Review A 70 (2004) 052328.
[28] V. Danos, E Kashefi, P. Panangaden, Parsimonious and robust realizations of unitary maps in the one-way model, Physical Review A 72 (2005) 064301.
[29] R. Raussendorf, D.E. Browne, H.J. Briegel, Measurement-based quantum computation on cluster states, Physical Review A 68 (2003) 022312.
[30] H. Vollmer, Introduction to Circuit Complexity, Springer-Verlag, 1999.
[31] M. Hein, J. Eisert, H.J. Briegel, Multi-party entanglement in graph states, Physical Review A 69 (2004) 062311.
[32] N. de Beaudrap, Finding flows in the one-way measurement model, Phys. Rev. A 77 (2008) 022328.
Available as http://arxiv.org/abs/quant-ph/0611284v2, 2006.
[33] N. de Beaudrap, V. Danos, E. Kashefi, Phase map decomposition for unitaries. Available as http://arxiv.org/abs/quant-ph/0603266v1, 2006.
[34] M. Mhalla, S. Perdrix, Complexity of graph state preparation. Available as http://arxiv.org/abs/quant-ph/0412071v1, 2004.
[35] R. Diestel, Graph Theory, Springer-Verlag, 2005.
[36] M. Furst, J.B. Saxe, M. Sipser, Parity, circuits, and the polynomial-time hierarchy, Theory of Computing Systems 17 (1984) 13–27.
[37] F. Verstraete, J.I. Cirac, Valence-bond states for quantum computation, Physical Review A 70 (2004) 060302(R).
[38] D.E. Browne, E. Kashefi, M. Mhalla, S. Perdrix, Generalized flow and determinism in measurement-based quantum computation, New Journal of Physics 9 (2007) 250.